

- [100] B. Baudry, S. Allier, and M. Monperrus, “Tailored source code transformations to synthesize computationally diverse program variants,” *ISSTA 2014*, p. 149–159, 2014.
- [101] M. Zalewski, “American Fuzzy Lop,” 2017.
- [102] K. Zhang, D. Wang, J. Xia, W. Y. Wang, and L. Li, “ALGO: Synthesizing Algorithmic Programs with Generated Oracle Verifiers,” *CoRR*, vol. abs/2305.14591, 2023.
- [103] L. de Moura and N. Bjørner, “Z3: An Efficient SMT Solver,” in *Tools and Algorithms for the Construction and Analysis of Systems*, (Berlin, Heidelberg), pp. 337–340, 2008.
- [104] A. Abate, C. David, P. Kesseli, D. Kroening, and E. Polgreen, “Counterexample Guided Inductive Synthesis Modulo Theories,” in *Proceedings of Computer Aided Verification - 30th International Conference, CAV*, vol. 10981, pp. 270–288, 2018.
- [105] P. M. Phothilimthana, A. Thakur, R. Bodik, and D. Dhurjati, “Scaling up Superoptimization,” in *Proceedings of the Twenty-First International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS*, pp. 297–310, 2016.
- [106] R. El-Khalil and A. D. Keromytis, “Hydan: Hiding Information in Program Binaries,” in *Information and Communications Security, 6th International Conference, ICICS*, vol. 3269, pp. 187–199, 2004.
- [107] V. Singhal, A. A. Pillai, C. Saumya, M. Kulkarni, and A. Machiry, “Cornucopia : A Framework for Feedback Guided Generation of Binaries,” in *37th IEEE/ACM International Conference on Automated Software Engineering, ASE 2022, Rochester, MI, USA, October 10-14, 2022*, pp. 27:1–27:13, ACM, 2022.
- [108] B. Cox and D. Evans, “N-Variant Systems: A Secretless Framework for Security through Diversity,” in *Proceedings of the 15th USENIX*, 2006.
- [109] D. Bruschi, L. Cavallaro, and A. Lanzi, “Diversified Process replicæ for Defeating Memory Error Exploits,” in *Proceedings of the 26th IEEE International Performance Computing and Communications Conference, IPCCC 2007, April 11-13, 2007, New Orleans, Louisiana, USA*, pp. 434–441, IEEE Computer Society, 2007.
- [110] B. Salamat, A. Gal, T. Jackson, K. Manivannan, G. Wagner, and M. Franz, “Stopping Buffer Overflow Attacks at Run-Time: Simultaneous Multi-variant Program Execution on a Multicore Processor,” tech. rep., Technical Report 07-13, School of Information and Computer Sciences, UC Irvine, 2007.

- [111] L. Davi, C. Liebchen, A. Sadeghi, K. Z. Snow, and F. Monrose, “Isomeron: Code Randomization Resilient to (Just-In-Time) Return-oriented Programming,” in *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*, The Internet Society, 2015.
- [112] G. Agosta, A. Barenghi, G. Pelosi, and M. Scandale, “The MEET Approach: Securing Cryptographic Embedded Software Against Side Channel Attacks,” *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 34, no. 8, pp. 1320–1333, 2015.
- [113] T. Jackson, B. Salamat, A. Homescu, K. Manivannan, G. Wagner, A. Gal, S. Brunthaler, C. Wimmer, and M. Franz, “Compiler-generated Software Diversity,” in *Moving Target Defense - Creating Asymmetric Uncertainty for Cyber Threats*, vol. 54, pp. 77–98, 2011.
- [114] A. Amarilli, S. Müller, D. Naccache, D. Page, P. Rauzy, and M. Tunstall, “Can Code Polymorphism Limit Information Leakage?,” in *Proceedings of Information Security Theory and Practice. Security and Privacy of Mobile Devices in Wireless Communication - 5th IFIP WG 11.2 International Workshop, WISTP*, vol. 6633, pp. 1–21, 2011.
- [115] A. Voulimeneas, D. Song, P. Larsen, M. Franz, and S. Volckaert, “dMVX: Secure and Efficient Multi-variant Execution in a Distributed Setting,” in *EuroSec ’21: Proceedings of the 14th European Workshop on Systems Security, Virtual Event / Edinburgh, Scotland, UK, April 26, 2021*, pp. 41–47, ACM, 2021.
- [116] V. Le, C. Sun, and Z. Su, “Finding Deep Compiler Bugs via Guided Stochastic Program Mutation,” in *Proceedings of the 2015 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications*, p. 386–399, 2015.
- [117] R. Tsoupidi, R. C. Lozano, and B. Baudry, “Constraint-based Diversification of JOP Gadgets,” *J. Artif. Intell. Res.*, vol. 72, pp. 1471–1505, 2021.
- [118] J. Falleri, F. Morandat, X. Blanc, M. Martinez, and M. Monperrus, “Fine-grained and Accurate Source Code Differencing,” in *ACM/IEEE International Conference on Automated Software Engineering, ASE ’14*, pp. 313–324, 2014.
- [119] S. Banescu, C. Collberg, and A. Pretschner, “Predicting the resilience of obfuscated code against symbolic execution attacks via machine learning,” in *26th USENIX Security Symposium (USENIX Security 17)*, pp. 661–678, Aug. 2017.